F²MC-16 Family

SOFTUNE™ V7
User's Manual

Software Support Manual

Publication Number MN704-00015 Revision 1.0 Issue Date August 24, 2015

Preface

Thank you for your continued use of Cypress semiconductor products.
Read the Hardware Manual and Data Sheet prior to using this product.

What is SOFTUNE

SOFTUNE Workbench is support software for developing programs of Cypress microcontrollers F^2MC-16 family.

It is a combination of a development manager, simulator debugger, emulator debugger,and an integrated development environment for efficient development.

Purpose of This Manual and Target Readers

This manual explains functions added / Modified / deleted to SOFTUNE V7 relative to SOFTUNE V3.

The manual is intended for engineers developing various types of products using SOFTUNE V7.

Trademark

SOFTUNE is the trademark of Cypress Semiconductor Corp..

The company names and brand names herein are the trademarks or registered trademarks of their respective owners.

Overall Organization of This Manual

This Manual has 5 chapters as shown below.

CHAPTER 1: SOFTUNE V7
CHAPTER 2: COMMAND REFERENCE MANUAL
CHAPTER 3: OPERATION MANUAL
CHAPTER 4: USER'S MANUAL
CHAPTER 5: APPENDIX

Related Manuals

The manuals related to SOFTUNE V7 are listed below. See the manual appropriate to the applicable conditions.

The contents of these manuals are subject to change without notice. Contact us to check the latest versions available.

SOFTUNE V7 Manual

- F^2MC-16 Family SOFTUNE V7 User's Manual (this manual)

SOFTUNE V3 Workbench Manual

- F^2MC-16 FAMILY SOFTUNE V3 Workbench COMMAND REFERENCE MANUAL (Called "COMMAND REFERENCE MANUAL" hereafter)
- F^2MC-16 FAMILY SOFTUNE V3 Workbench OPERATION MANUAL (Called "OPERATION MANUAL" hereafter)
- F^2MC-16 FAMILY SOFTUNE V3 Workbench USER'S MANUAL (Called "USER'S MANUAL" hereafter)

Language Tools Manual

- F^2MC-16 FAMILY SOFTUNE C COMPILER MANUAL for V3
- F^2MC-16 FAMILY SOFTUNE ASSEMBLER MANUAL for V3
- F^2MC-16 FAMILY SOFTUNE LINKAGE KIT MANUAL for V3

Hardware Manual

- F^2MC-16 Family HARDWARE MANUAL

The manuals for each series are provided.
See the appropriate manual for the series that you are using.

How to Use This Manual

Finding a Function

This manual explains functions added to / modified from / deleted from SOFTUNE V3 in SOFTUNE V7.If the function you wish to use is not given here, see the SOFTUNE V3 manual.

Terminology

This manual uses the following terminology.

Term	Explanation
SOFTUNE V7 tools	It is the tool group which constitutes SOFTUNE V7. It consists of the following tools. ・SOFTUNE V7 Workbench ・SOFTUNE V7 Language tools ・SOFTUNE USB Driver ・SOFTUNE V7 Common component
SOFTUNE V7 workbench	It is a microcontroller development environment platform. It consists of manager function and debugging functions.
Language tools	They are tools for translating a program and creating a load module. It consists of the following tools. ・C Compiler ・Assembler ・Linkage kit
SOFTUNE USB driver	It is a USB driver for SOFTUNE.
CPU family	This is the CPU supported in SOFTUNE.
Component	It is a file and the function which constitute a SOFTUNE V7 ProPack product.
V3/V6 format workspace	Workspace which does not include the information on the function added by SOFTUNE V7.
V7 format workspace	Workspace including the information on the function added by SOFTUNE V7.

Folder Names

This manual uses the following Folder names.

Folder Name	Explanation
The folder that installs	Top level folder specified when installing SOFTUNE V7.
CPU dependence folder	Contains the SOFTUNE Workbench corresponding to the CPU family. {The folder that installs}¥FMC16
Common folder	Contains files not dependent on the CPU family. {The folder that installs}¥Common
Workbench folders	Group of folders for SOFTUNE V7 Workbench. Folder in the CPU dependence folder, not the Lang folder. {CPU dependence folder}¥Bin {CPU dependence folder}¥Lib {CPU dependence folder}¥Sample
Lang folder	One or more Language tools are stored in this folder. Language tools are stored in subfolders (Language tools folders) for each version. {CPU dependence folder}¥Lang
Language tools folder	Folder directly below the Lang folder where the language tools for each version are stored. {Lang folder}¥REV700002¥ {Lang folder}¥REV7000xx¥
Selected language tools folder	Language tools folder of the version selected by the user.

Notations

■ This manual states chapter numbers and titles as follows.

− "x.y.z Title"

The chapter number is given as well as the chapter title of the manual being referred to.
(Example: "2.1.1 '1.1 Workspace Management Function'")

Table of Contents

CHAPTER 1: SOFTUNE V7 9
1. SOFTUNE V7 Overview 10
2. Function to Select Language Tools 11
2.1 To Select Language Tools When Creating a New Workspace / Project 11
2.1.1 When Creating a New Project File 11
2.1.2 When Creating a New Workspace 12
2.2 To Verify / Modify the Language Tools Selected 13
2.3 To Select Language Tools 14
3. Dispatch Function 15
4. Logging Build Log Function 18
5. Command Line Build Function 19
6. Other Modifications 20
CHAPTER 2: COMMAND REFERENCE MANUAL 21
1. Overview of Command Reference Manual Modifications 22
2. Explanation of Modifications 23
2.1 Modifications to"APPENDIX" 23
2.1.1 "APPENDIX A MANAGER-RELATED MESSAGES" 23
CHAPTER 3: OPERATION MANUAL 25
1. Overview of Operation Manual Modifications 26
2. Explanation of Modifications 27
2.1 Modifications to "CHAPTER 2 OPERATION" 28
2.1.1 "2.2.1 Startup Options" (Additions) 28
2.1.2 "2.2.2 Exit Code" (Additions) 29
2.1.3 "2.14 Reading SOFTUNE Project Files of Old Versions" 29
2.1.4 "2.16 Usual Functions" 29
2.2 Modifications to "CHAPTER 3 WINDOWS" 30
2.2.1 "3.6.1 Setting Standard Editor" 30
2.2.2 "3.16 Object Window" 30
2.2.3 "3.19 Sequence Window" 30
2.2.4 "3.20 Real-time Memory Window" 30
2.2.5 "3.21 RAM Checker Window" 30
2.3 Modifications to "CHAPTER 4 MENUS" - "4.1 Menu Configuration (Tree)" 31
2.4 Modifications to "CHAPTER 4 MENUS" - "4.2 File Menu" 32
2.4.1 "4.2.1 New" 32
2.4.2 "4.2.1.1 Select Language Tools Dialog" (Additions) 36
2.4.3 "4.2.4 Open Workspace" 39
2.4.4 "4.2.5 Close Workspace" 39
2.4.5 "4.2.7 Save As" 39
2.5 Modifications to "CHAPTER 4 MENUS" - "4.4 View Menu" 40
2.5.1 "4.4 View Menu" 40
2.5.2 "4.4.11 Object" 40
2.5.3 "4.4.15 Real-time Memory" 40
2.5.4 "4.4.16 RAM Checker" 40
2.6 Modifications to "CHAPTER 4 MENUS" - "4.5 Project" 41
2.6.1 "4.5.2.1 Add Project - New" 41
2.6.2 "4.5.2.2 Add Project - Existing Project" 42
2.6.3 "4.5.4 Setup Workspace" 43
2.6.4 "4.5.5.2 MCU" 45
2.6.5 "4.5.5.3 Setting C Compiler Options" 46
2.6.6 "4.5.6 Setting Customize Build" 47

2.6.7 "4.5.10 Compile, Make, Build, and Stop" ... 48
2.7 Modifications to "CHAPTER 4 MENUS" - "4.6 Debug" ... 49
2.7.1 "4.6 Debug" ... 49
2.7.2 "4.6.5 Event" ... 49
2.8 Modifications to "CHAPTER 4 MENUS" - "4.7 Setup" ... 50
2.8.1 "4.7.1 Development" ... 50
2.8.2 "4.7.2 Debug Environment" ... 53
2.8.3 "4.7.2.4 Selecting Debug Function" ... 53
2.9 Modifications to "CHAPTER 4 MENUS" - "4.9 Help" ... 54
2.9.1 "4.9 Help" ... 54
2.9.2 "4.9.1 Workbench / C Compiler / Assembler / Linkage Kit Help contents" ... 55
2.9.3 "4.9.2 Release note" ... 56
2.9.4 "4.9.3 About SOFTUNE" ... 57
2.10 Modifications to "CHAPTER 5 Add-in Module" ... 63
2.10.1 "5.1.1 What is Customize Bar?" ... 63
2.11 Modifications to "APPENDIX" ... 63
2.11.1 "APPENDIX C Setting LAN Interface" ... 63
2.11.2 "APPENDIX G Sample Project for Semihosting Function" ... 63
CHAPTER 4: USER'S MANUAL ... 65
1. Overview of USER'S MANUAL Modifications ... 66
2. Explanation of Modifications ... 67
2.1 Modifications to "CHAPTER1 Basic Functions" ... 67
2.1.1 "1.1 Workspace Management Function" ... 67
2.1.2 "1.2 Project Management Function" ... 67
2.1.3 "1.4 Make/Build Function" ... 67
2.1.4 "1.4.1 Customize Build Function" ... 67
2.1.5 "1.7 Error Jump Function" ... 67
2.1.6 "1.8 Editor Functions" ... 67
2.1.7 "1.12 Setting Operating Environment" ... 67
2.1.8 "1.13 Debugger Types" ... 68
2.2 Modifications to "CHAPTER2 Dependence Functions" ... 69
2.2.1 "2.1 Simulator Debugger" ... 69
2.2.2 "2.1.12 Checking Debugger Status" ... 69
2.2.3 "2.2 Emulator Debugger (MB2141)" ... 69
2.2.4 "2.3 Emulator Debugger (MB2147-01)" ... 69
2.2.5 "2.3 Emulator Debugger (MB2147-05)" ... 69
2.2.6 "2.5 Emulator Debugger (MB2198)" ... 69
2.2.7 "2.6.10 Checking Debugger Status" ... 69
2.2.8 "2.7 Monitor Debugger" ... 69
CHAPTER 5: APPENDIX ... 71
1. Major Changes ... 72

CHAPTER 1: SOFTUNE V7

This chapter explains SOFTUNE V7.

1. SOFTUNE V7 Overview
2. Function to Select Language Tools
3. Dispatch Function
4. Logging Build Log Function
5. Command Line Build Function
6. Other Modifications

1. SOFTUNE V7 Overview

There is a growing demand relating to product safety in software development. In accordance with this, there are the following requirements.

The desire to enable reproduction of development deliverables

- The desire to link workspace to language tools.

The desire to leave a record of development results

- The desire to save build results in a file.

To meet these requirements, the following functions were added to SOFTUNE V7 relative to SOFTUNE V3. This chapter describes these functions.

- Function to select language tools
  - You can select language tools to use in each workspace.
  - Use the language tools specified in workspace to build.
- Dispatch function
  - You can activate SOFTUNE with a saved workspace.
  - You can select / record SOFTUNE to use.
- Logging build log function
  - You can save logs during builds.
- Command line build function
  - You can build / make from command line.

In addition, the following points have been modified relative to SOFTUNE V3.

- You can make external tools operate before or after making or building.
- You can install several copies of SOFTUNE V7 on one PC.
- You can activate other than the user account which was installed.
- Only the following F^2MC-16 Family (F^2MC-16FX) debuggers are supported.
  - Emulator debugger (MB2100-01)
  - Simulator debugger

2. Function to Select Language Tools

You can select language tools to use in each workspace.

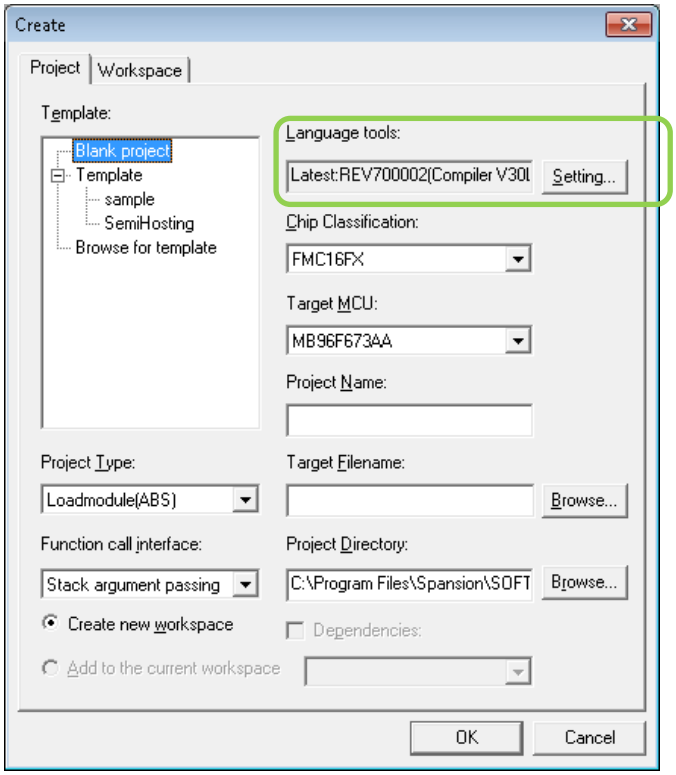
2.1 To Select Language Tools When Creating a New Workspace / Project

When creating a new workspace / project, the procedure to select language tools is as follows.

2.1.1 When Creating a New Project File

When creating a project in combination with a workspace, you can select from the [Language tools] box in the [Project] tab of the create dialog. Click [Setting] to open the select language tools dialog. For the select language tools, see "2.3 To Select Language Tools". The selected language tools are recorded in the workspace created together with the project.

Figure 2-1 When the [Project] Tab is Selected

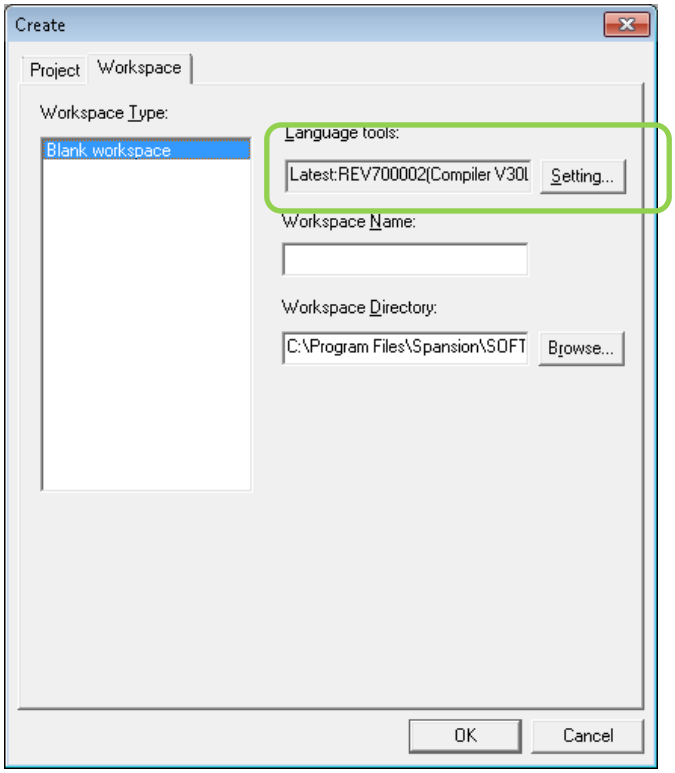


For details, see "CHAPTER 3" - '2.4.1"4.2.1 New"' in this manual.

2.1.2 When Creating a New Workspace

You can select in the [Language tools] box even if creating only a new workspace from the [Workspace] tab. Click [Setting] to open the select language tools dialog. For the select language tools, see "2.3 To Select Language Tools".

Figure 2-2 When the [Workspace] Tab is Selected

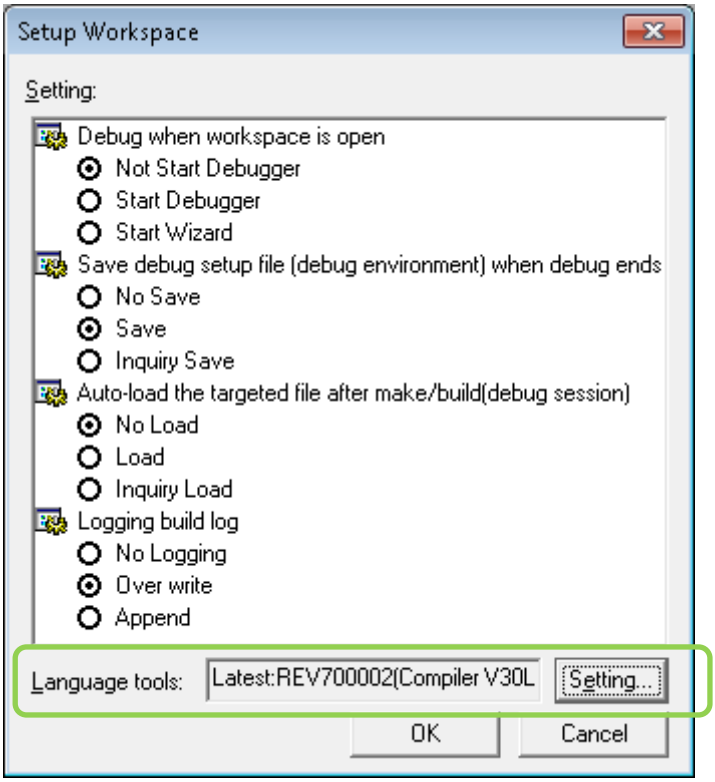


For details, see "CHAPTER 3" - '2.4.1"4.2.1 New"' in this manual.

2.2 To Verify / Modify the Language Tools Selected

You can verify what language tools are selected in the workspace with the setup workspace dialog. You can also make modifications from the select language tools dialog, the same as when creating a new workspace / project, by clicking [Setting]. For the select language tools, see "2.3 To Select Language Tools".

Figure 2-3 Setup Workspace Dialog

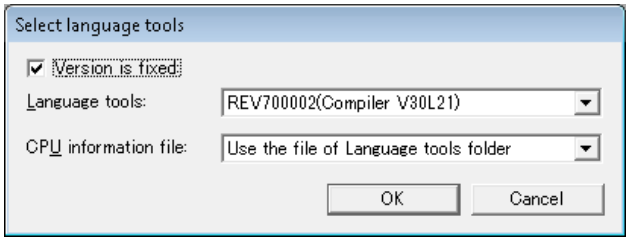


For details, see "CHAPTER 3" - '2.6.3"4.5.4 Setup Workspace"' in this manual.

2.3 To Select Language Tools

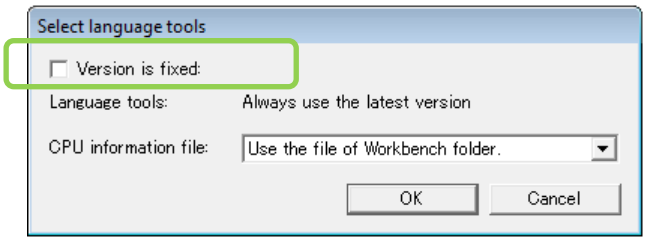
Click [Setting] in the create dialog [Project] or [Workspace] tabs, and in [Language tools] in the setup workspace dialog, to open the select language tools dialog and select language tools.

Figure 2-4 Select Language Tools Dialog (To Lock the Version)



If there is no need to lock the version, or if you wish to always use the latest version of the language tools installed, you can leave the version unlocked.

Figure 2-5 Select Language Tools Dialog (Leave the Version Unlocked)



For details, see "CHAPTER 3" - '2.4.2"4.2.1.1 Select Language Tools Dialog" (Additions)' in this manual.

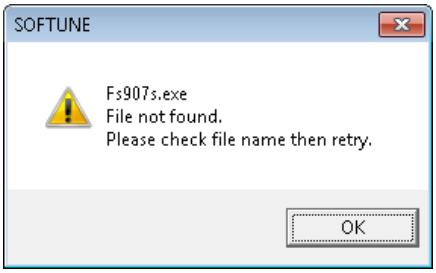
3. Dispatch Function

With the dispatch function, by double-clicking on a workspace you can directly activate the SOFTUNE Workbench which that workspace is saved for. (Realizes association of workspace file extension (.wsp) to Dispatcher executable file (SOFTUNE.exe))

Even if multiple copies of SOFTUNE Workbench are installed, the optimum SOFTUNE Workbench can still be used.

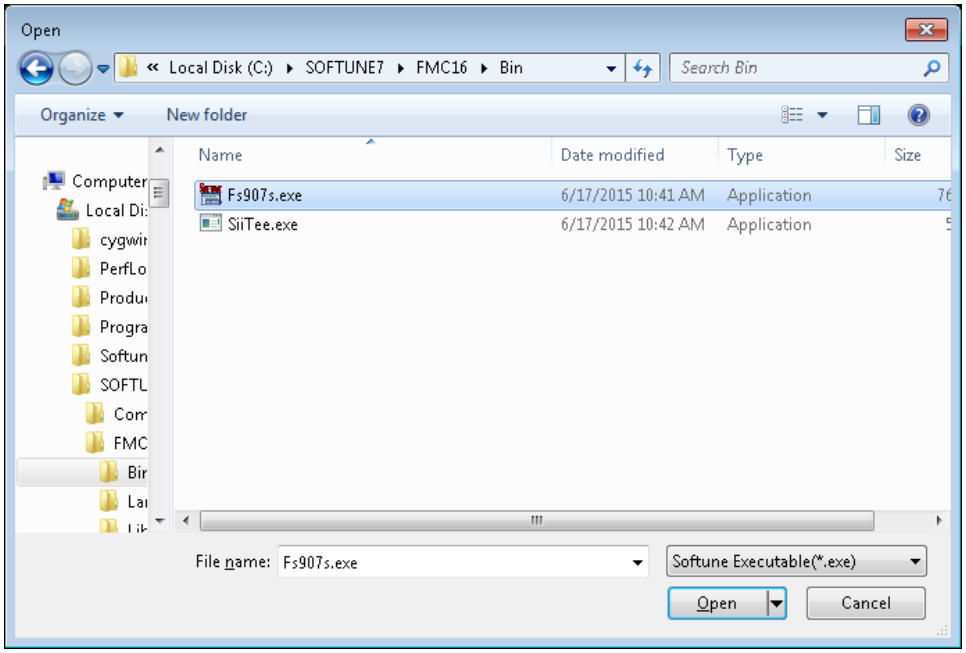
If the relevant SOFTUNE Workbench cannot be found, or in the case of workspace files created with SOFTUNE V3, the following dialog opens.

Figure 3-1 Warning Dialog (Dispatcher)



Click [OK] to open the following dialog, and you can select the SOFTUNE Workbench you wish to activate.

Figure 3-2 File Selection Dialog (Dispatcher)



- Open

Selecting the SOFTUNE Workbench executable file you wish to activate and clicking [Open] records the selected results in the workspace. The selected SOFTUNE Workbench is activated. If the workspace file has a read only attribute set to it, causing an error, remove the read only attribute.

- Cancel

Click [Cancel] to finish without activating SOFTUNE Workbench.

Note:

If the workspace file is not associated with a Dispatcher executable file:

You can associate it through one of following methods.

- "Open with"

Users can set this for each account.

Can be set even for user accounts without administrator authority.

- "Assoc / ftype command"

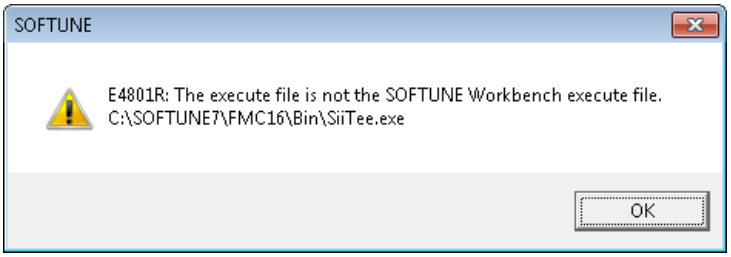
Sets for user accounts with administrator authority.

Valid for all user accounts on the PC which has been set.

■ Error Dialog

In some cases Dispatcher may display a warning message or error message.

Figure 3-3 Error Dialog (Dispatcher)



The displayed messages are as follows.

E4209R	Open file read error.

[Explanation] An error occurred during reading of the opened file.

[Operator response] Check the file (drive) being read.

E4302R	Failed during start. (executable file name)

[Explanation] Unable to activate tool.

[Operator response] The tool name may be false. Review the tool settings.

E4801R	The executable file is not the SOFTUNE Workbench executable file. (executable file name)

[Explanation] The executable file specified for SOFTUNE Workbench is not SOFTUNE Workbench.

[Operator response] Specify the correct SOFTUNE Workbench.

F9801R	Please invoke from SOFTUNE MultiVersionManager.

[Explanation] SOFTUNEV3/V6 was activated by Dispatcher in an environment where SOFTUNE MultiVersionManager is installed.

[Operator response] Activate from SOFTUNE MultiVersionManager in an environment where SOFTUNE MultiVersionManager is installed.

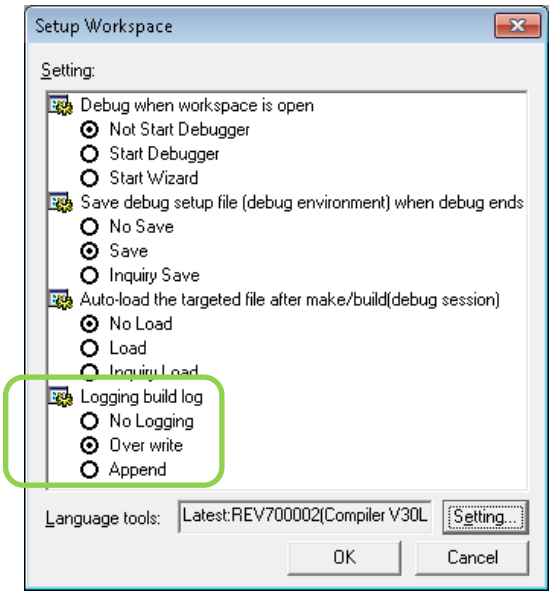
4. Logging Build Log Function

How to set

You can keep a log during builds. You can set this for each workspace.

Set the logging build log item.

Figure 4-1 Setup Workspace Dialog



For details, see "CHAPTER 3" - '2.6.3"4.5.4 Setup Workspace"' in this manual.

About build log

Build log is saved to the list file directory of the output directory.

File name will be " (project name).log".

The following items are saved.

Table 4-1 Explanation of Build Log

Items	Explanation
Header	The version and time stamp (*1) of SOFTUNE Workbench
	Language tools folder name
	The version of the language tools currently used
Log	Contents which language tools output to an output window when building (*2)

*1: A time stamp is saved only when "Over write " is selected.

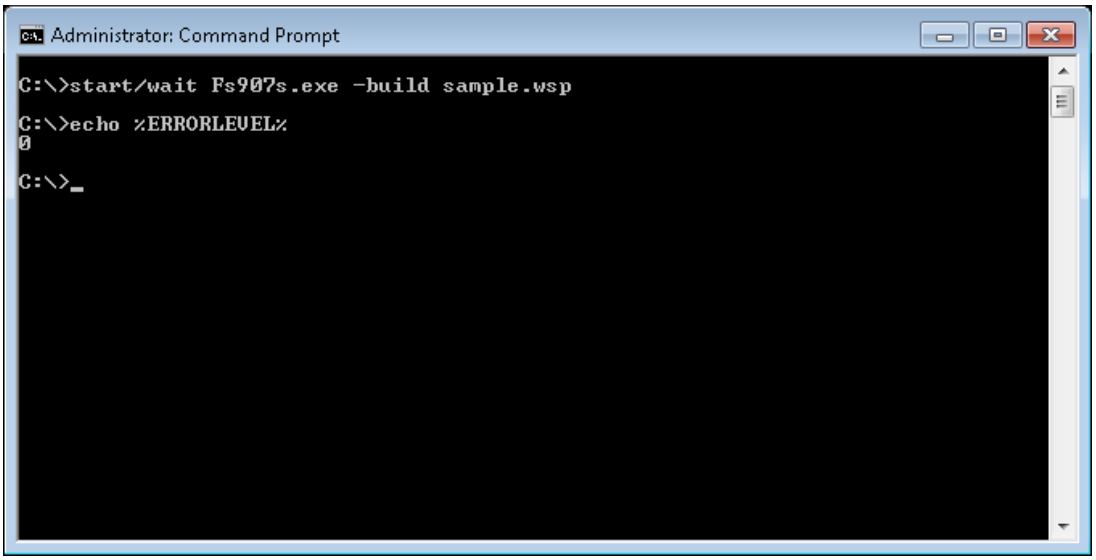
*2: If sub-projects are contained, the outputted contents when building sub-projects are also saved.

5. Command Line Build Function

With SOFTUNE Workbench, you can execute build and make functions from the command line.

Using these functions you can incorporate SOFTUNE Workbench into your build tools, etc.

Figure 5-1 Command Line Build



For details, see "CHAPTER 3"-'2.1.1"2.2.1 Startup Options" (Additions)' and '2.1.2"2.2.2 Exit Code" (Additions)' in this manual.

6. Other Modifications

SOFTUNE V7 has had the following points modified in comparison to SOFTUNE V3.

- You can make external tools operate before or after making or building.
  The custom build function was expanded.
  For details, see "CHAPTER 3"-'2.6.6"4.5.6 Setting Customize Build"' in this manual.

- You can install several copies of SOFTUNE V7 on one PC.
  See the Installation Manual.

- You can activate other than the user account it was installed with.

- Supports the F^2MC-16Family (F^2MC-16FX) debuggers.
  The following debuggers are not supported.
  - Emulator debugger (MB2198)
  - Emulator debugger (MB2147-01)

CHAPTER 2: COMMAND REFERENCE MANUAL

This chapter explains modifications to the SOFTUNE V7 Command Reference Manual.

1. Overview of Command Reference Manual Modifications
2. Explanation of Modifications

1. Overview of Command Reference Manual Modifications

The following points are modified in SOFTUNE V7 in the Command Reference Manual.

For more details, see "2. Explanation of Modifications".

Functions That Were Added

- You can select language tools.

Functions That Were Modified

- Supported debugger

2. Explanation of Modifications

The following debuggers are not supported. The explanation of the following debuggers does not apply to SOFTUNE V7.

- Normal simulator debugger (normal)
- Emulator debugger (MB2141)
- Emulator debugger (MB2147-01)
- Emulator debugger (MB2147-05)
- Emulator debugger (MB2198)
- Monitor debugger

For areas not explained in this manual, see the SOFTUNE V3 Workbench Command Reference Manual.

2.1 Modifications to"APPENDIX"

2.1.1 "APPENDIX A MANAGER-RELATED MESSAGES"

The following error messages were added.

E4406W	The target MCU specified in the project is not included in the CPU information file.

[Explanation] The target MCU specified in the project is not included in the CPU information file.

[Operator response] The following may be the cause.

When a language tool is chosen accidentally.

Please return to the select language tools dialog and choose the right language tools.

When a CPU information file is not appropriate.

Please end Workbench and replace to the correct CPU information file.

When the target MCU used in a project is not suitable.

Please change the Target MCU in a MCU change dialog.

E4407W	The target MCU used in the specified template project is not included in the CPU information file. (CPU Information file)

[Explanation] The target MCU used in the specified template project is not included in the CPU information file.

[Operator response] The following may be the cause.

When a language tool is chosen accidentally.

Please return to the select language tools dialog and choose the right language tools.

When a CPU information file is not appropriate.

Please end Workbench and replace to the correct CPU information file.

When the selected template project is incorrect.

Please re-select a template project.

E4800W	Following language tools folder is not found.(folder name)

[Explanation] The language tools folder specified in the workspace file cannot be found.

[Operator response] Install the language tools or click [OK] to modify the language tools.

E4801W	The compiler of the following version is not found at <folder-name>.(folder name)

[Explanation] Compiler of the specified version cannot be found in the folder specified in the workspace file.

[Operator response] Install the language tools or click [OK] to modify the language tools.

E4802W	This project file is too old format. It can not be read.

[Explanation] The project file is created by old SOFTUNE. The workspace file created by SOFTUNE of the following versions cannot be read by SOFTUNE V7.

FR:V50L04 and previous versions
FMC16:V30L24 and previous versions
FMC8:V30L24 and previous versions

[Operator response] Please read by SOFTUNE V7 after changing by reading by SOFTUNE of the following versions.

FR:V50L05 or later versions, V60L01 or later versions
FMC16:V30L25 or later versions
FMC8:V30L25 or later versions

E4803W	A language tools folder is not found.

[Explanation] The CPU information file of a language tools folder is used for the opened workspace.

However, a language tools folder is not found.

[Operator response] Please install a language tools. Or please click the [OK] button and use the CPU information file of a Workbench folder.

CHAPTER 3: OPERATION MANUAL

This chapter explains modifications to the SOFTUNE V7 OPERATION MANUAL.

1. Overview of Operation Manual Modifications
2. Explanation of Modifications

1. Overview of Operation Manual Modifications

The following points are modified in SOFTUNE V7 in the OPERATION MANUAL.

For more details, see "2. Explanation of Modifications".

Functions That Were Added

- You can select language tools.
- You can save the build log.

Functions That Were Modified

- Supported debugger
- Supported project format
- Customize build function
- CPU information file specification method
- Unique function in unsupported debugger
- Handling environment variables
- Display method for help / release files
- Format of workspace / project files
- Management method for standard editor settings
- Install folder configuration

2. Explanation of Modifications

The following debuggers are not supported. The explanation of the following debuggers does not apply to SOFTUNE V7.

- Normal simulator debugger (normal)
- Emulator debugger (MB2141)
- Emulator debugger (MB2147-01)
- Emulator debugger (MB2147-05)
- Emulator debugger (MB2198)
- Monitor debugger

The REALOS is not supported. The explanation of the REALOS does not apply to SOFTUNE V7.

This section explains the modifications to pages which are altered. For parts not explained in this manual, see the corresponding page in the SOFTUNE V3 Workbench Operation Manual.

2.1 Modifications to "CHAPTER 2 OPERATION"

These are the explanation of the modifications to "SOFTUNE V3 Workbench OPERATION MANUAL" - "CHAPTER 2 OPERATION."

2.1.1 "2.2.1 Startup Options" (Additions)

When starting up, SOFTUNE Workbench can execute the processes you specify with qualifiers and parameters.

Format

Fs907s.exe [qualifiers] [parameters]
[ ] : optional

■ qualifiers

Qualifiers are uppercase and lowercase-sensitive.

-make

After starting SOFTUNE Workbench, it executes make and then terminates SOFTUNE Workbench.
When starting SOFTUNE Workbench, it executes make using the Active Project / Configuration.
The make result is returned as an exit code.
If the workspace file name is not specified, it will only start SOFTUNE Workbench.
About make function, see "SOFTUNE V3 Workbench OPERATION MANUAL" - "2.11 Make/Build"

-build

After starting SOFTUNE Workbench, it executes build and then terminates SOFTUNE Workbench.
When starting SOFTUNE Workbench, it executes build using the Active Project / Configuration.
The build result is returned as an exit code.
If the workspace file name is not specified, it will only start SOFTUNE Workbench.
About build function, see "SOFTUNE V3 Workbench OPERATION MANUAL" - "2.11 Make/Build"

■ parameters

Workspace file name

Specify the workspace file to load when starting SOFTUNE Workbench.
The extension cannot be omitted.

Examle

If sample.wsp was built

> Fs907s.exe -build sample.wsp

2.1.2 "2.2.2 Exit Code" (Additions)

When SOFTUNE Workbench terminates, it returns an exit code. If no qualifiers were specified, it will return 0.

■ If the -make / -build qualifiers were specified

The following exit code is returned for the build result.

Table 2-1Exit Code

Exit Code	Description
-1	The tool can not start
0	normal end
1	warning
2	error end
3	fatal error end

Examle

If sample.wsp was built and it terminated normally

```
> start /wait Fs907s.exe -build sample.wsp
> echo %ERRORLEVEL%
0
```

2.1.3 "2.14 Reading SOFTUNE Project Files of Old Versions"

Project files created in the following versions of SOFTUNE Workbench cannot be read by SOFTUNE V7.

FMC16:V30L24 and previous versions

In the same way, project files created in SOFTUNE Manager V01/V02 cannot be read by SOFTUNE V7.

2.1.4 "2.16 Usual Functions"

The following window is not displayed. The explanation of this window does not apply to SOFTUNE V7.

- Real-time memory
- RAM Checker

2.2 Modifications to "CHAPTER 3 WINDOWS"

These are the explanation of the modifications to "SOFTUNE V3 Workbench OPERATION MANUAL" - "CHAPTER 3 WINDOWS."

2.2.1 "3.6.1 Setting Standard Editor"

The standard editor settings are now saved in the user folder file. In addition, the setting file for the standard editor can be browsed from multiple SOFTUNE V7 Workbenches. Accordingly, the settings of the standard editor will be shared for each user.

2.2.2 "3.16 Object Window"

This window is not displayed in SOFTUNE V7. The explanation in this section does not apply to SOFTUNE V7.

2.2.3 "3.19 Sequence Window"

This window is not displayed in SOFTUNE V7. The explanation in this section does not apply to SOFTUNE V7.

2.2.4 "3.20 Real-time Memory Window"

This window is not displayed in SOFTUNE V7. The explanation in this section does not apply to SOFTUNE V7.

2.2.5 "3.21 RAM Checker Window"

This window is not displayed in SOFTUNE V7. The explanation in this section does not apply to SOFTUNE V7.

2.3 Modifications to "CHAPTER 4 MENUS" - "4.1 Menu Configuration (Tree)"

The following menus are not displayed.

View

- Object
- Realtime Memory
- RAM checker
- Sequence

Debug

- Event

2.4 Modifications to "CHAPTER 4 MENUS" - "4.2 File Menu"

These are the explanation of the modifications to "SOFTUNE V3 Workbench OPERATION MANUAL" - "CHAPTER 4 MENUS" - "4.2 File Menu."

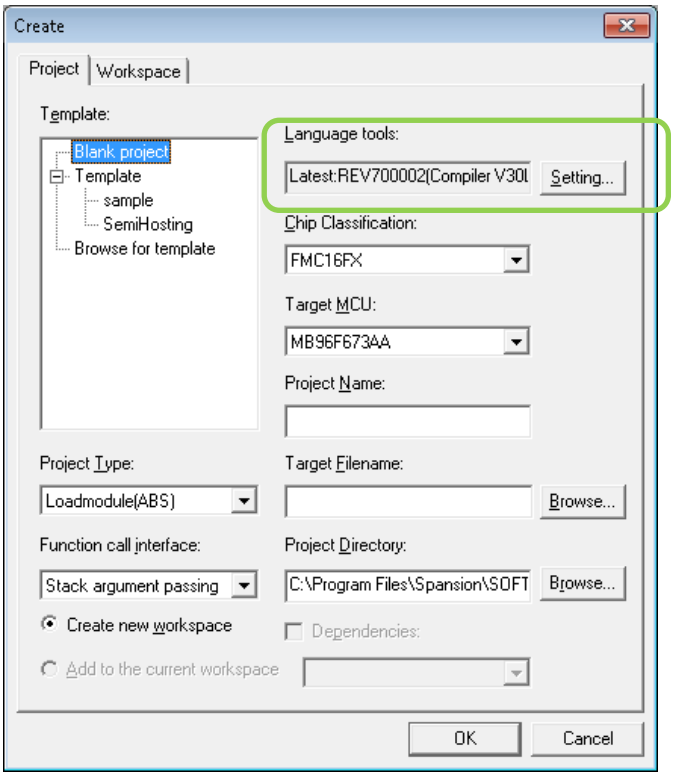
2.4.1 "4.2.1 New"

You can specify the language tools to use in the workspace. The respective dialogs are modified.

New Dialog

■ When [Workspace/Project file] is selected

Figure 2-1 When the [Project] Tab is Selected



- Language tools

Displays the name of the language tools folder to use in the workspace created simultaneously with the project, and the compiler version.

Placing the mouse cursor on the edit box displays the entire text via a tooltip.

This information is saved in a workspace file.

- If the language tools version is fixed:
  <Latest compiler folder name>(Compiler VxxLxx)
- If the language tools version is not fixed:
  Latest:< Latest compiler folder name>(Compiler VxxLxx)
  (If there are no language tools in the Lang folder or its subfolders, or in the SOFTUNE V3 installation folder, [Latest:None] is displayed.)

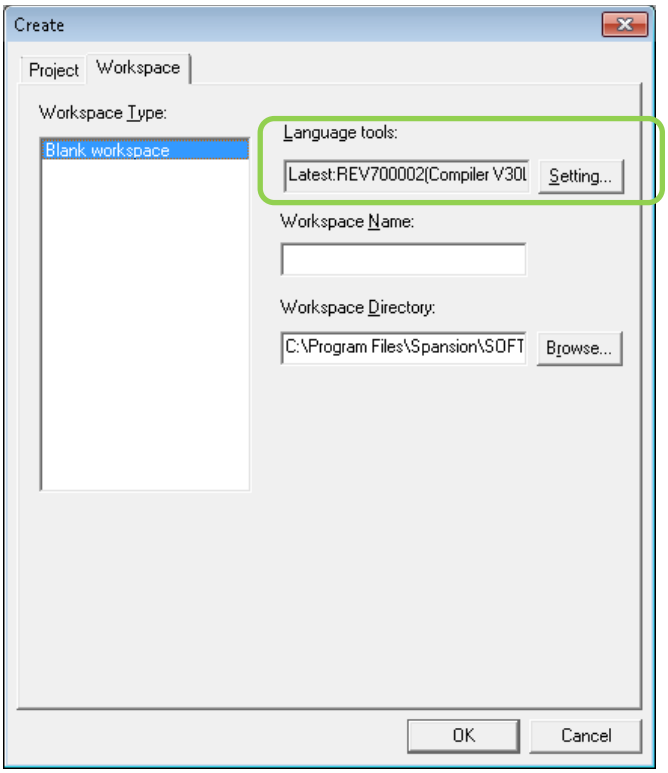
- Setting

You can modify language tools. The select language tools dialog is displayed.

For details, see '2.4.2 "4.2.1.1 Select Language Tools Dialog" (Additions) ' in this manual.

Modifying language tools causes [Chip type] and [Target MCU] to revert to initial values.

Figure 2-2 When the [Workspace] Tab is Selected



- Language tools

Displays the name of the language tools folder to use in the workspace, and the compiler version.

Placing the mouse cursor on the edit box displays the entire text via a tooltip.

- If the language tools version is fixed:
  <Latest compiler folder name>(Compiler VxxLxx)
- If the language tools version is not fixed:
  Latest:< Latest compiler folder name>(Compiler VxxLxx)
  (If there are no language tools in the Lang folder or its subfolders, or in the SOFTUNE V3 installation folder, [Latest:None] is displayed.)

- Setting

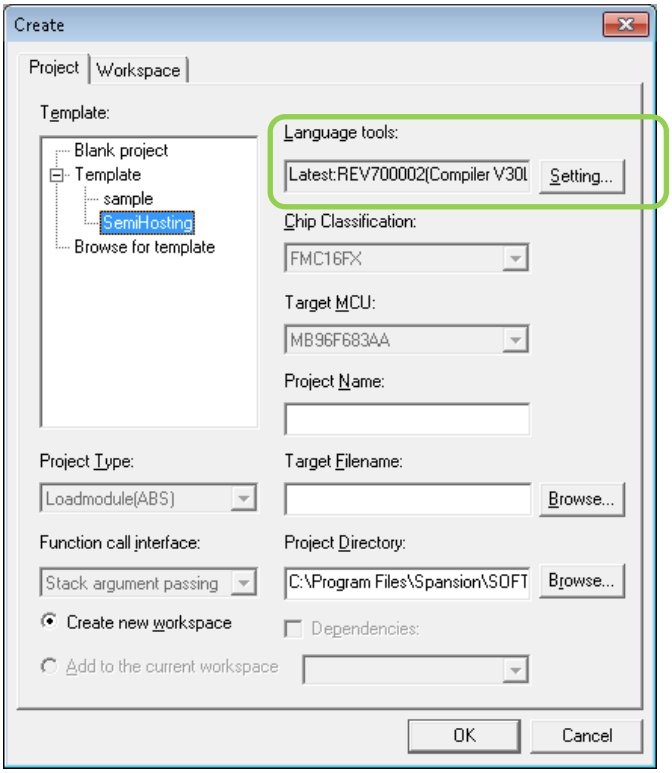
You can modify language tools. The select language tools dialog is displayed.

For details, see '2.4.2 "4.2.1.1 Select Language Tools Dialog" (Additions) ' in this manual.

Procedure for Creating a Workspace or Project

■ Creating a new project based on a template project

Figure 2-3 When [Template] is Selected



- Language tools

Displays the name of the language tools folder to use in the workspace created simultaneously with the project, and the compiler version.

Placing the mouse cursor on the edit box displays the entire text via a tooltip.

This information is saved in a workspace file.

- If the language tools version is fixed:
  <Latest compiler folder name>(Compiler VxxLxx)
- If the language tools version is not fixed:
  Latest:< Latest compiler folder name>(Compiler VxxLxx)
  (If there are no language tools in the Lang folder or its subfolders, or in the SOFTUNE V3 installation folder, [Latest:None] is displayed.)

- Setting

You can modify language tools. The select language tools dialog is displayed.

For details, see '2.4.2 "4.2.1.1 Select Language Tools Dialog" (Additions) ' in this manual.

- OK

A dialog is displayed to select members to create a project based on the template project.

Notes:

- *If the template is selected, neither [Chip Classification] nor [Target MCU] can be modified. The [Chip Classification] and [Target MCU] set in the template project are displayed. It is impossible to return to initial values even if language tools are modified.*
- *If a template project is used, there is the possibility that the selected target MCU might not exist in the CPU information file to use in the workspace. An error message is displayed in such a case. A warning message is displayed if the template project used is a project file in V3/V6 format.*

2.4.2 "4.2.1.1 Select Language Tools Dialog" (Additions)

A function to select language tools to use in the workspace was added.

In the Select language tools dialog, select the language tools to use in the workspace.

Select Language Tools Dialog

Click the language tools [Setting] button to display the select language tools dialog.

■ Select Language Tools Dialog

Select the language tools to use in the workspace.

Figure 2-4 Select Language Tools Dialog (Version is Fixed)

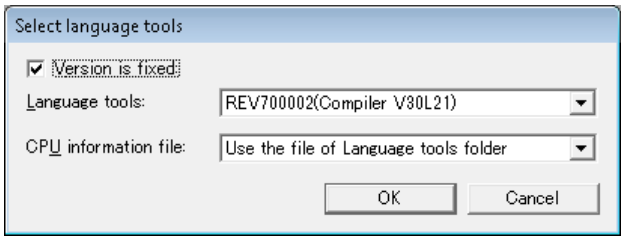
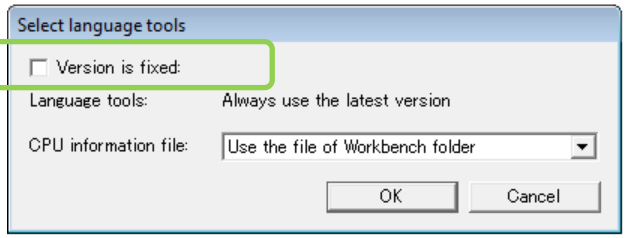


Figure 2-5 Select Language Tools Dialog (Version is Not Fixed)



- Version is fixed

You can fix the language tools version to use in the build by enabling the checkbox.

To fix the version, select the version to use from the [Language tools] box.

If the version is not fixed, [Always use the latest version] will apply.

Enabling/disabling the checkbox modifies the [CPU information file] box. Reset as necessary.

- Language tools
  - If the version is fixed, you can select the language tools to use from the list.
    The list displays the name in the language tools folder in the Lang folder, and the compiler version. It is sorted in order of the latest compiler version. The installation folder of SOFTUNE V3 of the same CPU family is displayed next.
    Placing the mouse cursor on the combo box displays the entire text via a tooltip.
    The language tools list is created when Workbench is launched. Restart Workbench if you added or deleted language tools.
  - If the version is not fixed, the latest language tools are always used. You cannot select language tools.
    Determine whether the language tool is new or old via the compiler version. Language tools in the folder containing the latest compiler inside the Lang folder are used. If there are no language tools in the Lang folder or its subfolders, the language tools in the SOFTUNE V3 installation folder are used.

- CPU Information file

Select the CPU information file to use in the workspace in the folder stored in that.

- Use the file of Language tools folder.
  The CPU information file in the selected language tools folder is used.
  The CPU information file suitable for the version of the selected language tool can be used.
- Use the file of Workbench folder.
  Whatever language tools are selected, the CPU information file in the "¥Lib" of the CPU dependency folder is used.
  Because CPU information files other than in the language tools folder are used, the following options are attached when compiling / making / building.
  -cif (CPU dependence folder)¥Lib¥[911|907|896].csv

- OK

Save the selection results and close the dialog.

If the language tools to be used are modified, a confirmation dialog is displayed.

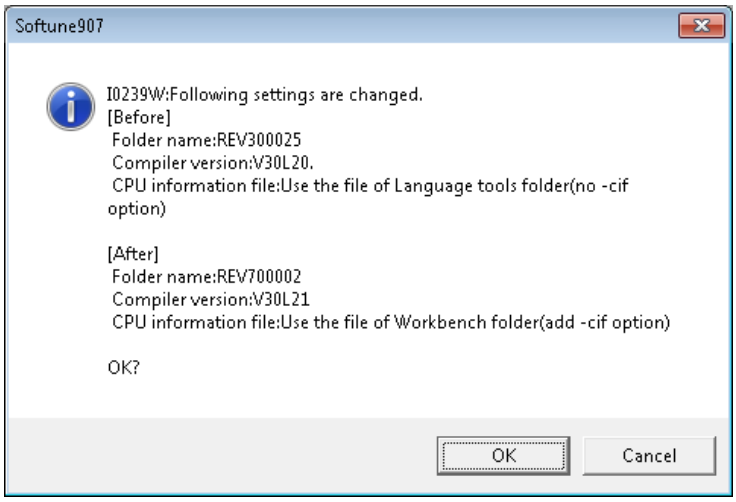
- Cancel

Discard the selection results and close the dialog.

■ Message to confirm modifications to language tools to use

If the language tools to use are modified in select language tools, a message to confirm whether to accept the modifications is displayed.

Figure 2-6 Message to Confirm Modifications to Language Tools



Click [OK] to save the selection results. The language tools to use are modified.

Click [Cancel] not to save the selection results and return to the select language tools dialog.

2.4.3 "4.2.4 Open Workspace"

The following notes are added.

Note:

The project file created by FR family SOFTUNE of V30L24 and previous versions cannot be read by SOFTUNE V7. Please read by SOFTUNE V7 after changing by reading with FMC16 family SOFTUNE of V30L25 or later versions.

If you open a workspace created in SOFTUNE V3, a dialog is displayed to confirm that the workspace file and project file formats will be modified. In addition, in SOFTUNE V7, a user cannot specify a CPU information file. When you push the [OK] button and you continue processing, see '2.4.2 "4.2.1.1 Select Language Tools Dialog" (Additions)'. Select the folder in which the CPU information file is stored.

[OK] button:	Continues the process to open the workspace.
[Cancel] button:	Cancels the process to open the workspace.

Moreover, when the language tool specified in the workspace file is not found, the dialog which checks whether another language tool is specified is displayed.

[OK] button:	The selection dialog of a language tool opens. The item changed here is recorded on a work space file, when saving work space.
[Cancellation] button:	The processing which opens work space is canceled.

2.4.4 "4.2.5 Close Workspace"

The conditions changed which open the dialog box for asking whether to save the current workspace information.

When the currently open workspace file is not edited at all

Also in the following cases, the dialog box for asking does not open:

- When workspace is opened by the executable file of SOFTUNE Workbench currently recorded on workspace file

2.4.5 "4.2.7 Save As"

You cannot select "Old project format (*.prj)" in the save formats for project files selectable in the [File types] combo box.

2.5 Modifications to "CHAPTER 4 MENUS" - "4.4 View Menu"

These are the explanation of the modifications to "SOFTUNE V3 Workbench OPERATION MANUAL " - "CHAPTER 4 MENUS" - "4.4 View Menu."

2.5.1 "4.4 View Menu"

The following windows are not displayed. The explanation of the respective windows does not apply to SOFTUNE V7.

- Object
- Realtime memory
- RAM Checker
- Sequence

2.5.2 "4.4.11 Object"

This window is not displayed in SOFTUNE V7. The explanation in this section does not apply to SOFTUNE V7.

- Object

2.5.3 "4.4.15 Real-time Memory"

This window is not displayed in SOFTUNE V7. The explanation in this section does not apply to SOFTUNE V7.

- Realtime memory

2.5.4 "4.4.16 RAM Checker"

This window is not displayed in SOFTUNE V7. The explanation in this section does not apply to SOFTUNE V7.

- RAM Checker

2.6 Modifications to "CHAPTER 4 MENUS" - "4.5 Project"

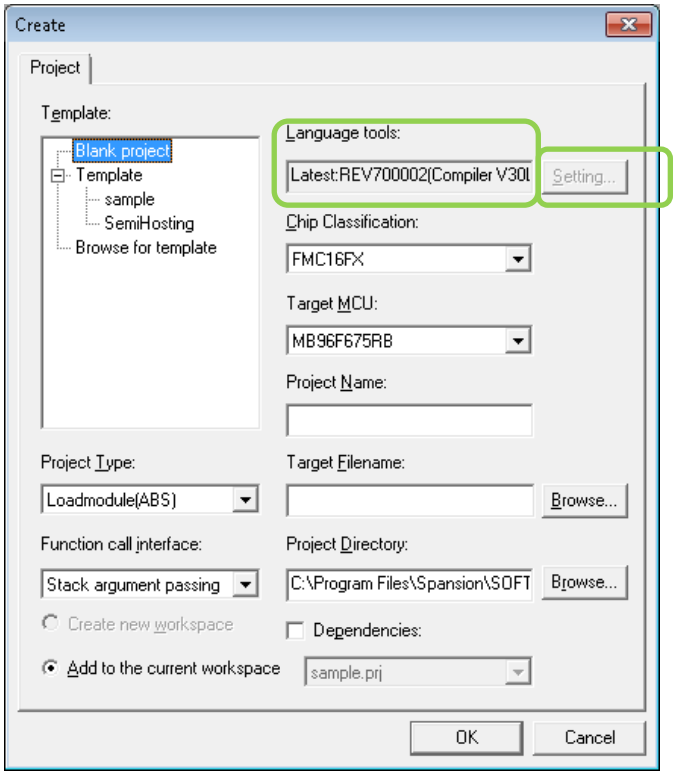
These are the explanation of the modifications to "SOFTUNE V3 Workbench OPERATION MANUAL" - "CHAPTER 4 MENUS" - "4.5 Project."

2.6.1 "4.5.2.1 Add Project - New"

A function to display language tools set in the workspace was added.

[REALOS (ABS)] cannot be selected in "Project Type".

Figure 2-7 Dialog for Creating Projects



- Language tools

If you add a project to the current workspace, the language tools set for the current workspace are displayed. Modification is not allowed.

Placing the mouse cursor on the edit box displays the entire text via a tooltip.

- If the language tools version is fixed:
  <Latest compiler folder name>(Compiler VxxLxx)
- If the language tools version is not fixed:
  Latest:< Latest compiler folder name>(Compiler VxxLxx)
  (If there are no language tools in the Lang folder or its subfolders, or in the SOFTUNE V3 installation folder, [Latest:None] is displayed.)

- Setting

Language tools cannot be changed in this dialog. See '2.6.3 "4.5.4 Setup Workspace" ' dialog.

2.6.2 "4.5.2.2 Add Project - Existing Project"

The following notes are added.

Note:

If you open a project created in SOFTUNE V3, a dialog is displayed to confirm that the project file format will be modified.

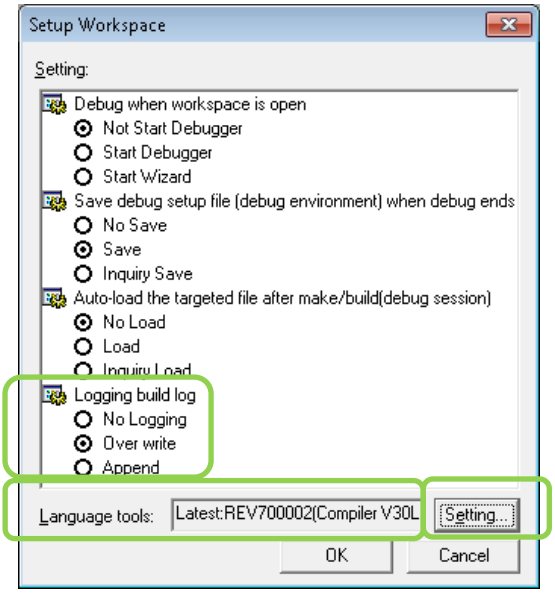
[OK] button:	Continues the process to add a project.
[Cancel] button:	Cancels the process to add a project.

2.6.3 "4.5.4 Setup Workspace"

The following functions are added.

- Build log logging function
- Language tools selecting function

Figure 2-8 Setup Workspace Dialog



- Logging build log

Specifies whether to save the build log. Does not interfere with the logging of the output window.

For details of Build log, See "CHAPTER1" - '4. Logging Build Log Function' in this manual.

- Language tools

The language tools to use in the current workspace are displayed.

- If the language tools version is fixed:
  <Latest compiler folder name>(Compiler VxxLxx)
- If the language tools version is not fixed:
  Latest:< Latest compiler folder name>(Compiler VxxLxx)
  (If there are no language tools in the Lang folder or its subfolders, or in the SOFTUNE V3 installation folder, [Latest:None] is displayed.)

- Setting

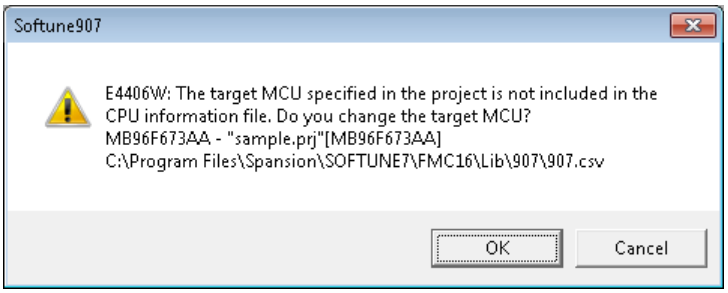
The select language tools dialog is displayed.
For details, see '2.4.2 "4.2.1.1 Select Language Tools Dialog" (Additions) ' in this manual.

During debugging, language tools cannot be selected. [Setting] button becomes invalid.

- OK

If the language tools are modified, it is possible that the target MCU to use in the project might not exist in the CPU information file to use. An error message is displayed in such a case.

Figure 2-9 Message Stating That the Target MCU Does Not Exist in the CPU Information File


Softune907
E4406W: The target MCU specified in the project is not included in the CPU information file. Do you change the target MCU?
MB96F673AA - "sample.prj"[MB96F673AA]
C:\Program Files\Spansion\SOFTUNE7\FMC16\Lib\907\907.csv
OK
Cancel

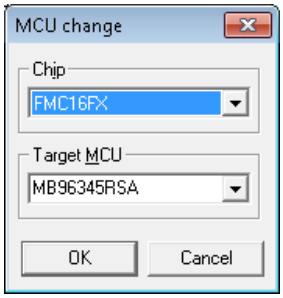
Click [OK] to display the MCU change dialog.

Click [Cance] to close the dialog and return to the previous dialog.

■ MCU change dialog

Selects chip and target MCU.

Figure 2-10 MCU Change Dialog



− OK

Save the selection results. The type of chip to use and the target MCU are modified.

− Cancel

Close the dialog without saving the selection results.

2.6.4 "4.5.5.2 MCU"

The following functions are deleted.

- [MCU change]

  A function to borrow the settings of another MCU to use a target MCU not defined in the CPU information file. The CPU information file corresponding to the latest target MCU is publicly available, so use the appropriate CPU information file.

- [Set CPU Information]

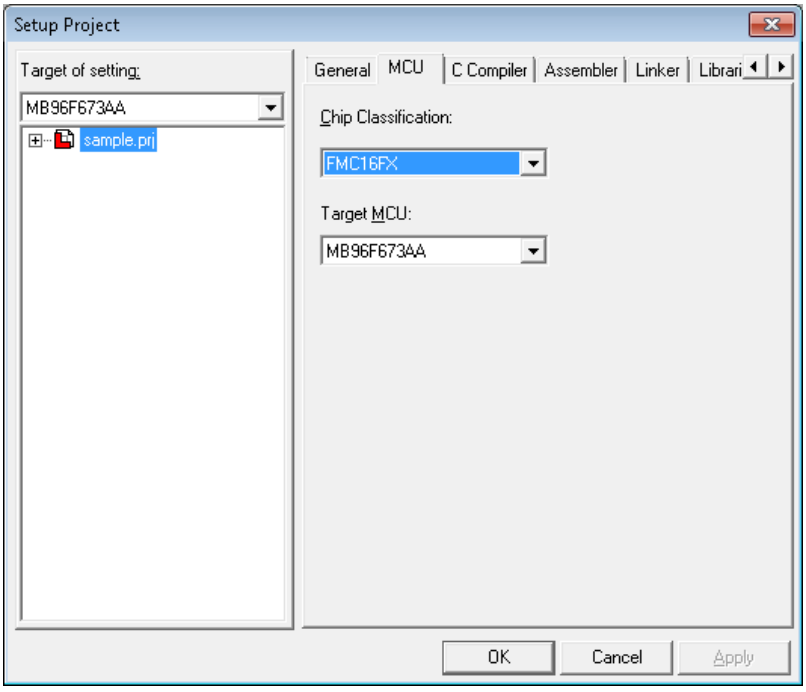
  A function to switch the CPU information file to use. Multiple language tools can be used in SOFTUNE V7. Use by associating the appropriate CPU information file corresponding to the language tools version.

- [Set CPU Execute Mode]

  A function to set emulators not supported in SOFTUNE V7.

Setting of MCU

Figure 2-11 Setup Project Dialog - MCU

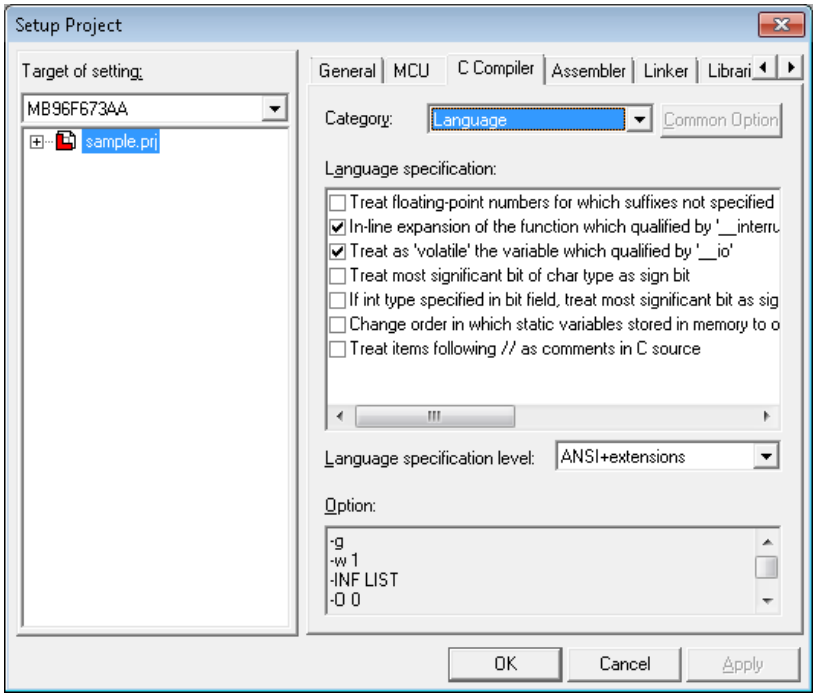


2.6.5 "4.5.5.3 Setting C Compiler Options"

The REALOS is not supported in SOFTUNE V7. The "In-line of ITRON system call development" item to set this function was deleted from the following dialog's [Language] box.

Setting of Options Included in [Language]

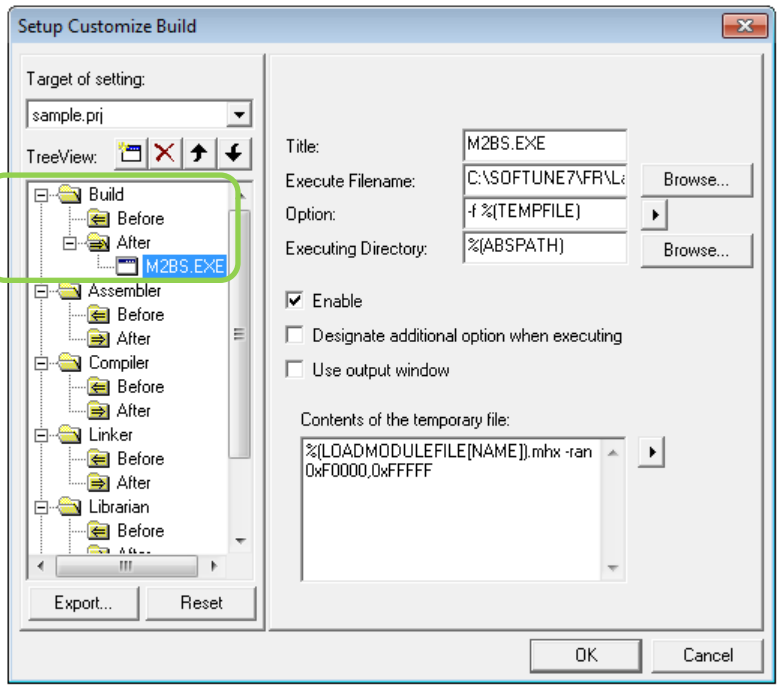
Figure 2-12Language Specification Option Setup Dialog



2.6.6 "4.5.6 Setting Customize Build"

Other tools from previous / later makes or builds can now be made to work, and are added to the categories selectable on the tree view.

Figure 2-13 Setup Customize Build Dialog Box



- About [Build]-[Before], [Build]-[After]

The [Build]-[Before] and [Build]-[After] categories execute the first and last build processing(*) sequence.

Even if a sub-project is contained, it does not execute before and after the project build. The [Build]-[Before] and [Build]-[After] category error check differs from the language tools [Before] and [After] category error check, and it will be immediately closing if there is a [Build]-[Before] processing error. The error check is only carried out if the [Use output window] checkbox is enabled.

*: "Build processing" refers to the following operations.

- [Project] menu - [Make]
- [Project] menu - [Build]
- [Tool bar] - [Make] button
- [Tool bar] - [Build] button
- [Project] window - pop-up menu - [Make]
- [Project] window - pop-up menu - [Build]
- Execute make by specifying the -make qualifier in the startup options
- Execute build by specifying the -build qualifier in the startup options

For [Compile] in the [Project] window pop-up menu, the [Build]-[Before], [Build]-[After] categories are not executed.

2.6.7 "4.5.10 Compile, Make, Build, and Stop"

Use the language tools selected in the workspace to compile, make and build. See "CHAPTER 1" - "2. Function to Select Language Tools" in this manual for the language tools used for compiling, making and building.

A log is kept when compiling, making and building. See "CHAPTER 1" - "4. Logging Build Log Function" in this manual for the build log.

2.7 Modifications to "CHAPTER 4 MENUS" - "4.6 Debug"

These are the explanation of the modifications to "SOFTUNE V3 Workbench OPERATION MANUAL" - "CHAPTER 4 MENUS" - "4.6 Debug."

2.7.1 "4.6 Debug"

The following menu is not displayed. Set the performance interval from the performance window. See "SOFTUNE V3 Workbench OPERATION MANUAL" - "4.4.13.4 Performance (Emulator Debugger [MB2100-01])"

- Event

2.7.2 "4.6.5 Event"

Set the performance interval from the performance window. See "SOFTUNE V3 Workbench OPERATION MANUAL" - "4.4.13.4 Performance (Emulator Debugger [MB2100-01])"

2.8 Modifications to "CHAPTER 4 MENUS" - "4.7 Setup"

These are the explanation of the modifications to "SOFTUNE V3 Workbench OPERATION MANUAL" - "CHAPTER 4 MENUS" - "4.7 Setup."

2.8.1 "4.7.1 Development"

Some of the environment variables are generated by Workbench based on the language tools folder selected in the workspace. The user cannot modify the ones generated by Workbench.

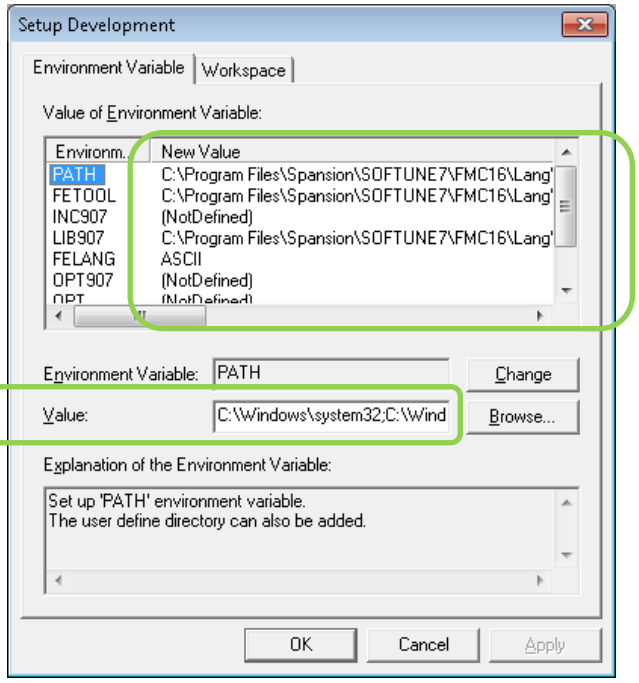
Environment variable

Table 2-2 Environment Variable Names

Environment variable name	Explanation
FETOOL	It is the selected language tool folder. It is the folder of the base in which the language tool used for building is installed. Modification is not allowed. Undefined if workspace is not open.
INC907	Folder to make searches of standard header files (*1).
PATH	Folder to make searches of executable files. If a workspace is open, Workbench adds the following folders. − (Selected language tools folder)¥Bin
LIB907	Folder to make searches of library files. If a workspace is open, Workbench adds the following folders. − (Selected language tools folder)¥Lib¥907
FELANG	Switches the character code of the messages output by language tools. (SJIS: Japanese (shift JIS), ASCII: English)
OPT907	Folder storing the default option file for the compiler, assembler and linker (*1).
OPT	Folder storing the default option file for the librarian and object format converter (*1).
TMP	Folder to create a temporary work file.

*1: This is a default environment variable and is not defined. In such a case, compiler or assembler browses for the appropriate folder based on the environment variable FETOOL. Accordingly, the default status behavior does not differ from SOFTUNE V3 with values generated by FETOOL set as the defaults.

Figure 2-14 Development Environment Setup (Environment Variable) Dialog Box



Here the environment variable settings shown in Table 2-2 are applied.

- [Value of Environment Variable] list - [New Value]

  If a workspace is not open or if language tools are not installed, only the values set for [Value] are displayed.

- [Value]

  Only the values of the parts set by users are displayed in the [Value] edit box. [Value] has settings shared by all SOFTUNEs in the user account.

Set the development environment in the following procedure:

1. Select the name of the environment variable whose setting is to be changed from the [Value of Environment Variable] list.
   - The content of [New Value] is displayed in the [Value] field.
   - Simple explanation of the environment variable is displayed in [Explanation of the Environment Variable].
2. Change the description of the [Value] field.

   It can be modified by the following two methods.

   - Inputting manually

     Input the folder name manually in the [Value] field. You can input multiple folder names by separating with semicolon.

   - Inputting using the Browse button

     Click the Browse button to display the "Browse Folders" dialog. The folder name selected on the dialog is added to the [Value] field. To input multiple folder names, repeat this steps as many times as needed.
3. Click the [Change] button.

If the setting value is left blank, "(Not Defined)" is displayed for the [New Value] field.

Note:

- *Environment variables are saved according to each user account, so even if build is executed with the same project/same language tools, you may get a different build result.*

2.8.2 "4.7.2 Debug Environment"

The following item is not displayed. The explanation of the following item does not apply to SOFTUNE V7.

- Selecting Debug Function

2.8.3 "4.7.2.4 Selecting Debug Function"

The following dialogs is not displayed. The explanation of the following dialog does not apply to SOFTUNE V7.

- Select debug function

2.9 Modifications to "CHAPTER 4 MENUS" - "4.9 Help"

These are the explanation of the modifications to "SOFTUNE V3 Workbench OPERATION MANUAL" - "CHAPTER 4 MENUS" - "4.9 Help."

2.9.1 "4.9 Help"

The following items are displayed.

- Workbench help contents
- C Compiler help contents
- Assembler help contents
- Linkage Kit help contents
- Release note
- About SOFTUNE

2.9.2 "4.9.1 Workbench / C Compiler / Assembler / Linkage Kit Help contents"

The menu items are modified. Substitute "SOFTUNE V3 Workbench OPERATION MANUAL" - "4.9.1 Help Topics" with the following contents.

Help for the SOFTUNE Tools is displayed.

Workbench help contents
Displays Workbench help.

C Compiler help contents
Displays help for C compiler in the language tools folder selected in the workspace. You cannot select if the workspace is not open.

Assembler help contents
Displays help for Assembler in the language tools folder selected in the workspace. You cannot select if the workspace is not open.

Linkage Kit help contents
Displays help for Linkage kit in the language tools folder selected in the workspace. You cannot select if the workspace is not open.

2.9.3 "4.9.2 Release note"

The menu items are modified. Substitute "SOFTUNE V3 Workbench OPERATION MANUAL" - "4.9.2 Support Information " with the following contents.

The release notes are displayed.

Release note

The release notes contain information not included in the attached manual.

We recommend reading them prior to using SOFTUNE Workbench.

- Workbench
  Displays Workbench release notes.

- C Compiler
  Displays release notes for C compiler in the language tools folder selected in the workspace. You cannot select if the workspace is not open.

- Assembler Pack
  Displays release notes for Assembler pack in the language tools folder selected in the workspace. You cannot select if the workspace is not open.

2.9.4 "4.9.3 About SOFTUNE"

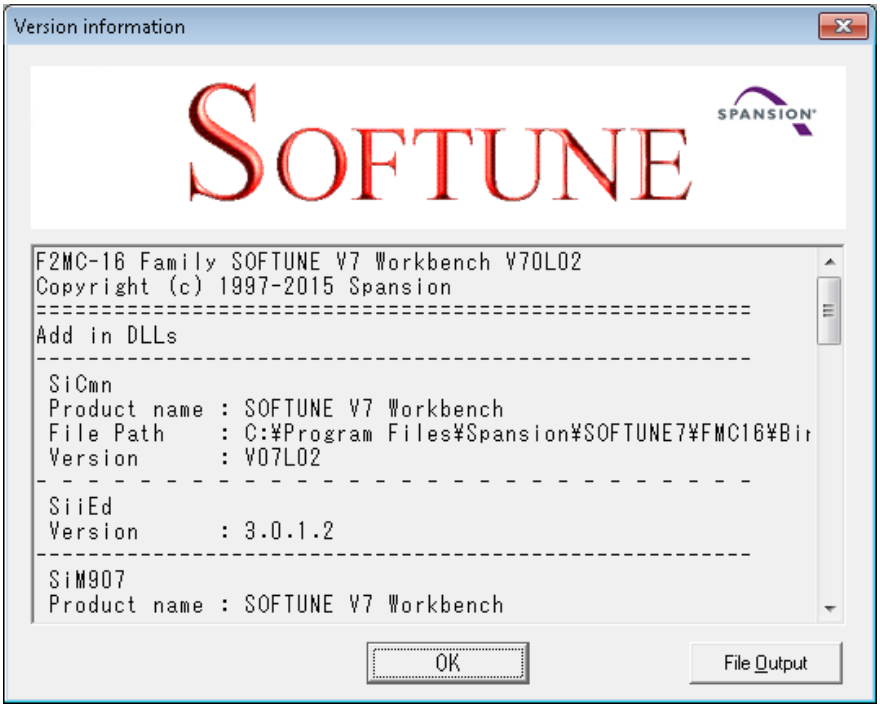
The contents displayed in the version information dialog have been modified.

The SOFTUNE's version information is displayed.

When [Help] - [About SOFTUNE] menu is selected, the version information dialog with the SOFTUNE Workbench logo is displayed.

This dialog displays each file version of SOFTUNE Workbench currently being activated.

Figure 2-15 Version Information



Version Information Overview

The version information dialog displays the following information.

1. SOFTUNE Workbench version and copyright information
2. Version and paths of add-in DLL loaded for SOFTUNE
3. Paths for current directory and help for SOFTUNE

■ SOFTUNE Workbench version and copyright information

The following information is displayed.

F^2MC-16 Family SOFTUNE V7 Workbench V70L02
Copyright (c) 1997-2015 Spansion

■ Version and paths of add-in DLL loaded for SOFTUNE.

If workspace is open and debugging has started, the information like the following (given as an example) is displayed. If workspace is not open and debugging is not started, some items are not displayed.

Information on DLL not installed is not displayed. The display order is subject to change.

■ Example) SiCmn

```
=====================================================
Add in DLLs
-----------------------------------------------------
 SiCmn
 Product name : SOFTUNE V7 Workbench
 File Path    : C:¥Program Files¥Spansion¥SOFTUNE7¥FMC16¥Bin¥SiC907.dll
 Version      : V07L02
- - - - - - - - - - - - - - - - - - - - - - - - - - - -
 SiiEd
 Version      : 3.0.1.2
```

■ Example) SiM907 (display if workspace is not open)

```
-----------------------------------------------------
 SiM907
 Product name : SOFTUNE V7 Workbench
 File Path    : C:¥Program Files¥Spansion¥SOFTUNE7¥FMC16¥Bin¥SiM907.dll
 Version      : V07L7002
- - - - - - - - - - - - - - - - - - - - - - - - - - - -
 Selectable Language Tools
  REV700002(Compiler V30L21)
  REV300025(Compiler V30L20.)
```

■ **Example) SiM907 (display if workspace is open)**

```
-------------------------------------------------------
 SiM907
 Product name : SOFTUNE V7 Workbench
 File Path    : C:¥Program Files¥Spansion¥SOFTUNE7¥FMC16¥Bin¥SiM907.dll
 Version      : V07L7002
- - - - - - - - - - - - - - - - - - - - - - - - - - - -
 Selectable Language Tools
  REV700002(Compiler V30L21)
  REV300025(Compiler V30L20.)
- - - - - - - - - - - - - - - - - - - - - - - - - - - -
 CPU Information file(Language tools folder)
  File Path :C:¥Program Files¥Spansion¥SOFTUNE7¥FMC16¥Lang¥REV700002¥Lib¥907¥907.csv
  Version   :13,33,01_2013.11.26
- - - - - - - - - - - - - - - - - - - - - - - - - - - -
 Language Tools(REV700002)
 - F2MC-16 Family SOFTUNE C Compiler V30L21
   File Path  : C:¥PROGRA~1¥Spansion¥SOFTUNE7¥FMC16¥Lang¥REV700~1¥Bin¥fcc907s.exe
 - F2MC-16 Family SOFTUNE Assembler V30L17
   File Path  : C:¥PROGRA~1¥Spansion¥SOFTUNE7¥FMC16¥Lang¥REV700~1¥Bin¥fasm907s.exe
 - F2MC-16 Family SOFTUNE Linker V30L16
   File Path  : C:¥PROGRA~1¥Spansion¥SOFTUNE7¥FMC16¥Lang¥REV700~1¥Bin¥flnk907s.exe
 - F2MC-16 Family SOFTUNE Librarian V30L08
   File Path  : C:¥PROGRA~1¥Spansion¥SOFTUNE7¥FMC16¥Lang¥REV700~1¥Bin¥flib907s.exe
 - SOFTUNE-OMF to S-FORMAT Converter V30L09
   File Path  : C:¥PROGRA~1¥Spansion¥SOFTUNE7¥FMC16¥Lang¥REV700~1¥Bin¥f2ms.exe
 - SOFTUNE-OMF to INTEL-HEX Converter V30L09
   File Path  : C:¥PROGRA~1¥Spansion¥SOFTUNE7¥FMC16¥Lang¥REV700~1¥Bin¥f2is.exe
 - SOFTUNE-OMF to INTEL-EXT-HEX Converter V30L09
   File Path  : C:¥PROGRA~1¥Spansion¥SOFTUNE7¥FMC16¥Lang¥REV700~1¥Bin¥f2es.exe
 - SOFTUNE-OMF to HEX Converter V30L09
   File Path  : C:¥PROGRA~1¥Spansion¥SOFTUNE7¥FMC16¥Lang¥REV700~1¥Bin¥f2hs.exe
```

■ **Example) SiD907 (displayed if debugging has stared)**

```
-------------------------------------------------------
 F2MC-16 Family Debugger DLL
 Product name : SOFTUNE V7 Workbench
 File Path    : C:¥Program Files¥Spansion¥SOFTUNE7¥FMC16¥Bin¥SiD907.dll
 Version      : V07L02
- - - - - - - - - - - - - - - - - - - - - - - - - - - - -
Debugger type       = Emulator Debugger
MCU type           = MB96F675RA
VCpu dll name       = C:¥Program Files¥Spansion¥SOFTUNE7¥FMC16¥Bin¥wv907e5.dll
VCpu dll version    = V07L02
SiDRVo dll version   = V04L00R05
Adapter version     = V01L05*(V02L04)
FPGA version        = V01L11
Maker ID           = 0x0400
CPU family ID       = 0x0100
DSU type ID         = 0x4
DSU version ID      = 0x1
Device ID          = 0x0022
Device version ID    = 0x1
OSC clock          = 1639892(1640000) Hz
PLL clock          = ---(---)
Clock mode          = RC
Communication mode   = Normal(Normal)
Communication type   = 0x1
Communication device = USB
```

SiD907 information differs according to each debugging as shown in Table 2-3.

Table 2-3 SiD907 Information

Item / Type	Description	Simulator debugger	Emulator debugger
			MB2100-01
Debugger type	Current debug type	○	○
MCU type	Currently selected target MCU	○	○
VCpu dll name	Path and name of VCpu dll currently being used	○	○
VCpu dll version	Version of VCpu dll currently being used	○	○
SiDRVo dll version	Version of driver dll currently being used	×	○
Adapter version	Adapter version	×	○
FPGA version	FPGA version	×	○
Maker ID	Device manufacturer ID	×	○
CPU family ID	ID of CPU family installed in device	×	○
DSU type ID	OCD-DSU mounting type ID	×	○
DSU version ID	ID of version information of DSU installed in device	×	○
Device ID	Device information ID	×	○
Device version ID	Device version information ID	×	○
OSC clock	Setting value (user setting) of reference frequency of standard communication speed	×	○
PLL clock	Setting value (user setting) of reference frequency of high communication speed	×	○
Clock mode	Clock mode	×	○
Communication mode	Communication mode during debug	×	○
Communication type	Initial communication mode [0: NRZ modulation / 1:phase modulation]	×	○
Communication device	Device type	×	○

Symbols in the table have the following meanings.

○ : Is displayed.

× : Is not displayed

■ **Example) SiIODef (The display order is subject to change)**

```
-------------------------------------------------------
 SiIODef
 Product name : SOFTUNE V7 Workbench
 File Path    : C:¥Program Files¥Spansion¥SOFTUNE7¥FMC16¥Bin¥SiIODef.dll
 Version      : V07L01
```

■ **Example) Customize Bar**

```
-------------------------------------------------------
 Custombar DLL
 Product name : SOFTUNE V7 Workbench
 File Path    : C:¥Program Files¥Spansion¥SOFTUNE7¥FMC16¥Bin¥SiTBar.dll
 Version      : V07L01
```

■ Paths for current directory and help for SOFTUNE

Information like the following examples is displayed. If workspace is not open, some items are not displayed.

■ **Example)**

```
=========================================================
Install Path   = C:¥Program Files¥Spansion¥SOFTUNE7¥FMC16
Current path   = C:¥Program Files¥Spansion¥SOFTUNE7¥FMC16¥Sample¥907¥Project
Language      = English
Help file path = C:¥Program Files¥Spansion¥SOFTUNE7¥FMC16¥LIB¥907¥SOFTUNE_HelpE.chm
              C:¥Program Files¥Spansion¥SOFTUNE7¥FMC16¥Lang¥REV700002¥LIB¥907¥FCCE.chm
              C:¥Program Files¥Spansion¥SOFTUNE7¥FMC16¥Lang¥REV700002¥LIB¥907¥FASME.chm
              C:¥Program Files¥Spansion¥SOFTUNE7¥FMC16¥Lang¥REV700002¥LIB¥907¥FLKITE.chm
```

File Output

Saves the data displayed in the version information dialog box into a file.

The default file name is fixed to "SOFTUNE_SYS.txt".

Reference:

When you contact us about SOFTUNE Workbench, please provide us with the information displayed in the above dialog box.

2.10 Modifications to "CHAPTER 5 Add-in Module"

These are the explanation of the modifications to "SOFTUNE V3 Workbench OPERATION MANUAL" - "CHAPTER 5 Add-in Module."

2.10.1 "5.1.1 What is Customize Bar?"

In SOFTUNE V7, customize bar is permanently installed. The explanation for "Installing Customize Bar" does not apply to SOFTUNE V7.

2.11 Modifications to "APPENDIX"

These are the explanation of the modifications to "SOFTUNE V3 Workbench OPERATION MANUAL" - "APPENDIX."

2.11.1 "APPENDIX C Setting LAN Interface"

Emulators with LAN interfaces are not supported. The explanation of LAN interfaces does not apply to SOFTUNE V7.

2.11.2 "APPENDIX G Sample Project for Semihosting Function"

Sample projects are stored in the following folder.

(The folder that installed SOFTUNE)¥FMC16¥Sample¥907¥Semihosting

CHAPTER 4: USER'S MANUAL

This chapter explains modifications to the SOFTUNE V7 User's Manual.

1. Overview of USER'S MANUAL Modifications
2. Explanation of Modifications

1. Overview of USER'S MANUAL Modifications

The following points are modified in SOFTUNE V7 in the USER'S MANUAL.

For more details, see the "2. Explanation of Modifications".

Functions that were added

- You can select language tools.

Functions that were modified

- Supported debugger
- Supported project format
- Unique function in unsupported debugger
- Format of workspace / project files
- Management method for standard editor settings

2. Explanation of Modifications

This section explains the modifications to pages which are altered. For parts not explained in this manual, see the corresponding page in the "SOFTUNE V3 Workbench USER'S MANUAL".

2.1 Modifications to "CHAPTER1 Basic Functions"

These are the explanation of the modifications to "SOFTUNE V3 Workbench USER'S MANUAL"-"CHAPTER 1 Basic Functions".

2.1.1 "1.1 Workspace Management Function"

The following function is not supported. The explanation of the following function does not apply to SOFTUNE V7.

- REALOS

In addition, the following function was added.

- Selection of language tools

2.1.2 "1.2 Project Management Function"

The following format is not supported. The explanation of this format does not apply to SOFTUNE V7.

- Old project format

2.1.3 "1.4 Make/Build Function"

Language tools used in making / building can be selected for each workspace. See "CHAPTER 3: OPERATING MANUAL"-'2.4.2"4.2.1.1 Select Language Tools Dialog" (Additions)' in this manual for how to select them.

2.1.4 "1.4.1 Customize Build Function"

In SOFTUNE V7, other tools can be automatically operated before or after the build.

2.1.5 "1.7 Error Jump Function"

In SOFTUNE V7 Workbench V70L01, a help file for SOFTUNE V3 is appended.

As a result, the error jump function does not activate for error messages added in SOFTUNE V7.

2.1.6 "1.8 Editor Functions"

The standard editor settings are now saved in the user folder file. In addition, the setting file for the standard editor can be browsed from multiple SOFTUNE V7 Workbenches. As a result, standard editor settings can be shared for each user.

2.1.7 "1.12 Setting Operating Environment"

Some of the environment variables are generated by Workbench based on the language tools folder selected in the workspace. The user cannot modify the ones generated by Workbench.

In addition, environment variables set in SOFTUNE V3 cannot be inherited in SOFTUNE V7. Register the appropriate settings as necessary.

2.1.8 "1.13 Debugger Types"

The following debuggers are not supported. The explanation of the respective debuggers does not apply to SOFTUNE V7.

- Monitor debugger

2.2 Modifications to "CHAPTER2 Dependence Functions"

These are the explanation of the modifications to "SOFTUNE V3 Workbench USER'S MANUAL"-"CHAPTER2 Dependence Functions".

2.2.1 "2.1 Simulator Debugger"

The following debugger is not supported. The explanation of the following debugger does not apply to SOFTUNE V7.

- Normal simulator debugger (normal)

2.2.2 "2.1.13 Checking Debugger Status"

See "CHAPTER3" - '2.9.4 "4.9.3 About SOFTUNE"' in this manual for the contents displayed for debugging information.

2.2.3 "2.2 Emulator Debugger (MB2141)"

The following debugger is not supported. The explanation of the following debugger does not apply to SOFTUNE V7.

- Emulator debugger (MB2141)

2.2.4 "2.3 Emulator Debugger (MB2147-01)"

The following debugger is not supported. The explanation of the following debugger does not apply to SOFTUNE V7.

- Emulator debugger (MB2147-01)

2.2.5 "2.4 Emulator Debugger (MB2147-05)"

The following debugger is not supported. The explanation of the following debugger does not apply to SOFTUNE V7.

- Emulator debugger (MB2147-05)

2.2.6 "2.5 Emulator Debugger (MB2198)"

The following debugger is not supported. The explanation of the following debugger does not apply to SOFTUNE V7.

- Emulator debugger (MB2198)

2.2.7 "2.6.10 Checking Debugger Status"

See "CHAPTER3" - '2.9.4 "4.9.3 About SOFTUNE" in this manual for the contents displayed for debugging information.

2.2.8 "2.7 Monitor Debugger"

The following debugger is not supported. The explanation of the following debugger does not apply to SOFTUNE V7.

- Monitor debugger

CHAPTER 5: APPENDIX

1. Major Changes

1. Major Changes

Page	Section	Change Results
Revision 1.0		
-	-	Initial release

MN704-00015-1v0-E

Cypress • Software Support Manual

F^{2}MC-16 Family
SOFTUNE V7 User's Manual

August 2015 Rev. 1.0

Published: Cypress Semiconductor Corp.
Edited: Communications

Colophon

The products described in this document are designed, developed and manufactured as contemplated for general use, including without limitation, ordinary industrial use, general office use, personal use, and household use, but are not designed, developed and manufactured as contemplated (1) for any use that includes fatal risks or dangers that, unless extremely high safety is secured, could have a serious effect to the public, and could lead directly to death, personal injury, severe physical damage or other loss (i.e., nuclear reaction control in nuclear facility, aircraft flight control, air traffic control, mass transport control, medical life support system, missile launch control in weapon system), or (2) for any use where chance of failure is intolerable (i.e., submersible repeater and artificial satellite). Please note that Cypress will not be liable to you and/or any third party for any claims or damages arising in connection with above-mentioned uses of the products. Any semiconductor devices have an inherent chance of failure. You must protect against injury, damage or loss from such failures by incorporating safety design measures into your facility and equipment such as redundancy, fire protection, and prevention of over-current levels and other abnormal operating conditions. If any products described in this document represent goods or technologies subject to certain restrictions on export under the Foreign Exchange and Foreign Trade Law of Japan, the US Export Administration Regulations or the applicable laws of any other country, the prior authorization by the respective government entity will be required for export of those products.

Trademarks and Notice

The contents of this document are subject to change without notice. This document may contain information on a Cypress product under development by Cypress. Cypress reserves the right to change or discontinue work on any product without notice. The information in this document is provided as is without warranty or guarantee of any kind as to its accuracy, completeness, operability, fitness for particular purpose, merchantability, non-infringement of third-party rights, or any other warranty, express, implied, or statutory. Cypress assumes no liability for any damages of any kind arising out of the use of the information in this document.

Copyright © 2015 Cypress Semiconductor Corp. All rights reserved. Cypress, the Cypress logo, Spansion®, the Spansion logo, MirrorBit®, MirrorBit® Eclipse™, ORNAND™, Easy DesignSim™, Traveo™ and combinations thereof, are trademarks and registered trademarks of Cypress Semiconductor Corp. in the United States and other countries. Other names used are for informational purposes only and may be trademarks of their respective owners.